MUCS@DravidianLangTech-2024: Role of Learning Approaches in Strengthening Hate-Alert Systems for code-mixed text

Manavi K K[a], **Sonali**[b], **Gauthamraj**[c]
Kavya G[d], **Asha Hegde** [e], **H L Shashirekha**[f]
Department of Computer Science, Mangalore University, Mangalore, Karnataka, India
{[a]kkmanavi, [b]sonalikulal417, [c]gauthamrajdataspace}@gmail.com,
{[d]kavyamujk, [e]hegdekasha}@gmail.com, [f]hlsrekha@mangaloreuniversity.ac.in

Abstract

Hate and Offensive (HOF) language detection is the task of detecting HOF content targeting a person or a group of people. Detecting HOF content is essential for promoting safety and positive engagement in online spaces, while also upholding community standards and protecting users from harm. However, despite massive efforts, it still remains challenging to effectively detect HOF content on online platforms because of ever-growing creative users. In view of this, to address the identification of HOF content on social media platforms, this paper describes the learning models submitted by our team - MUCS to "Hate and Offensive Language Detection in Telugu Codemixed Text (HOLD-Telugu): DravidianLangTech@EACL" - a shared task organized at European Chapter of the Association for Computational Linguistics (EACL) 2024. Three models: i) Logistic Regression (LR) model - a Machine Learning (ML) algorithm trained with Term Frequency-Inverse Document Frequency (TF-IDF) of character and word sequences in the range (1, 5) and (1, 3) respectively, and sub-words, ii) Ensemble model - a combination of ML classifiers (Multinomial Naive Bayes (MNB), LR, and Gaussian Naive Bayes (GNB)) trained with CountVectorizer of character and word sequences in the range (1, 5) and (1, 3), respectively, and iii) HateExplain_TL - a model based on Transfer Learning (TL) approach using Bidirectional Encoder Representations from Transformers (BERT) variant, are submitted to the shared task for detecting HOF content in Telugu code-mixed text. The proposed LR model outperformed the other models with a macro F1 score of 0.65.

1 Introduction

Twitter, Facebook, LinkedIn, Instagram, and other social media platforms have become popular places for people to spend their time and communicate with each other (Dikshitha Vani and Bharathi, 2022). While social media platforms offers numerous benefits, it also comes with drawbacks, including the spread of harmful content such as hate speech, offensive, abusive, and fake news content. Hate speech refers to any type of communication that targets, disparages, or encourages violence against an individual or group of people (Velankar et al., 2021).

Disseminating hateful content about a group or a community has a detrimental effect on those who are targeted by it. These victims experience stress, depression, and other mental health issues, and in extreme circumstances, they might even commit suicide (Roy et al., 2022). Therefore, it is necessary to detect HOF content to maintain healthy online platforms. Usually HOF content on social media is written by mixing words or sub-words belonging to more than one language known as code-mixed text. The code-mixed nature of HOF content is challenging because of its linguistic diversity (Priyadharshini et al., 2023b).

"Hate and Offensive Language Detection in Telugu Codemixed Text (HOLD-Telugu)" (B et al., 2024; Priyadharshini et al., 2023a), encourages the researchers to develop models to detect the HOF content in Telugu code-mixed texts. We - team MUCS, describe the three distinct models: i) LR model - a ML classifier fed with TF-IDF of character and word sequences in the range (1, 5) and (1, 3) respectively, and sub-words, ii) Ensemble model - a combination of ML classifiers (MNB, LR, and GNB) trained with CountVectorizer of character and word sequences in the range (1, 5) and (1, 3), respectively, and iii) HateExplain_TL - a model based on TL approach using BERT variant[1], for detecting HOF content in Telugu code-mixed texts.

The rest of the paper is organized as follows: while Section 2 describes the literature on HOF language identification in social media text, Sec-

[1] https://huggingface.co/Hate-speech-CNERG/bert-base-uncased-hatexplain

Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 252–256
March 22, 2024 ©2024 Association for Computational Linguistics

tion 3 focuses on the description of the models submitted to the shared task followed by the experiments and results in Section 4. Conclusion and future works are included in Section 5.

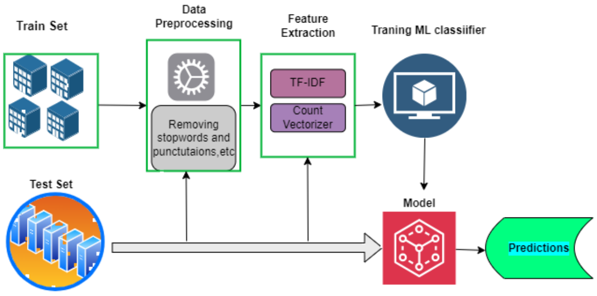


Figure 1: Framework of the proposed ML models

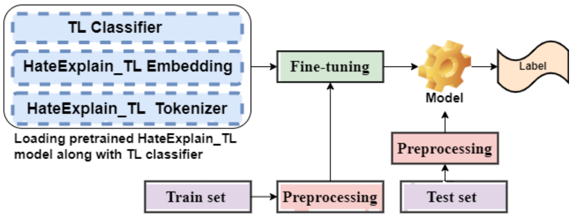


Figure 2: Framework of the proposed HateExplain_TL model

2 Related Work

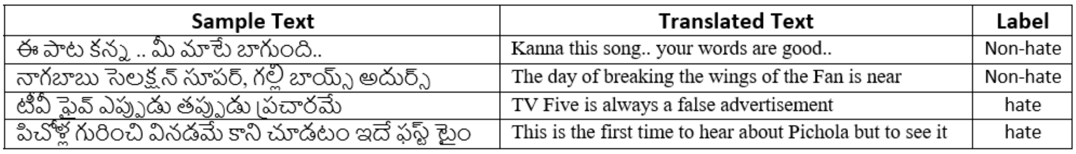
HOF content detection in code-mixed text is a growing area of study and several researchers have contributed to this area. Some of the related works for detecting HOF language are described below: To identify HOF content in Malayalam and Tamil code-mixed texts, Pathak et al. (2021) presented ML models (Support Vector Classifier (SVC), MNB, LR, and Random Forest (RF)) trained with TF-IDF of character and word sequences in the range (1, 5) and (1, 2) respectively for Malayalam code-mixed text, and TF-IDF of character and word sequences in the range (1, 7) and (1, 4) respectively for Tamil code-mixed text. They also trained ML models concatenating character and word TF-IDF. Among their proposed models, SVC models outperformed other models obtaining macro F1 scores of 0.74 and 0.86 for Malayalam and Tamil code-mixed texts respectively. Bhawal et al. (2021) experimented with ML (LR, RF, NB, eXtreme Gradient Boosting (XGBoost) and Support Vector Machine (SVM)), Deep Learning (DL) (Deep Neural Network (DNN), Convolutional Neural Network (CNN), and Bidirectional Long Short-Term Memory (Bi-LSTM)), and Transfer Learning (TL) (multilingual BERT (mBERT), a multilingual ALBERT model (IndicBERT), and Multilingual Representations for Indian Languages (MuRIL)) models, for HOF content detection in Malayalam and Tamil code-mixed texts. Their proposed ML and DL models were trained with TF-IDF of word n-grams in the range (1, 5) and for TL models, the corresponding BERT-based embeddings are used as features for training the classifiers. Out of their proposed models, the MuRIL model performed better with weighted F1 scores of 0.636 and 0.734 for Tamil and Malayalam code-mixed texts respectively.

Hegde et al. (2023) proposed two distinct models: i) AbusiveML - a Linear Support Vector Classifier (LinearSVC) trained with TF-IDF of word and character n-grams both in the range (1, 3) and ii) AbusiveTL - a model based on TL based approach with three BERT variants (Distilled Multilingual BERT (DistilmBERT), Tamil BERT, and Telugu BERT), for HOF content detection in Tamil, Telugu and romanized Tamil (RTamil) code-mixed texts. Their proposed AbusiveTL model outperformed the other models with macro F1 scores of 0.46, 0.74, and 0.49 securing 1st, 1st, and 4th ranks for code-mixed Tamil, Telugu, and RTamil texts respectively. Banerjee et al. (2021) fine-tuned various BERT models (mBERT-base, Cross-lingual Language Model with Robustly Optimized BERT approach (XLMR) - large, XLMR-base) on code-mixed Hindi texts and Hindi and English languages for binary (Non Hate-Offensive (NOT), HOF (HOF)) and multi-class (Hate speech (HATE), Offensive (OFFN), Profane (PRFN), Non-Hate (NONE)) tasks. Their proposed XLMR-large model obtained macro F1 scores of 0.7107, 0.8006, and 0.6447 for code-mixed Hindi (four classes), English (four classes), and English (two classes) texts respectively. Further, mBERT-base model obtained a macro F1 score of 0.7797 for Hindi (two classes) text.

From the above literature, it is found that there are several techniques for detecting HOF content in code-mixed text. However, there are only few studies that focus on Telugu code-mixed text indicating the need for further research and innovation in this field.

3 Methodology

To identify the HOF content in code-mixed Telugu text three distinct models: i) LR model ii) Ensemble model, and iii) HateExplain_TL models are pro-



Sample Text	Translated Text	Label
ఈ పాట కన్న .. మీ మాటే బాగుంది..	Kanna this song.. your words are good..	Non-hate
నాగబాబు సెలక్షన్ సూపర్, గల్లీ బాయ్స్ అదుర్స్	The day of breaking the wings of the Fan is near	Non-hate
టీవీ ఫైవ్ ఎప్పుడు తప్పుడు ప్రచారమే	TV Five is always a false advertisement	hate
పిచోళ్ల గురించి వినడమే కాని చూడటం ఇదే ఫస్ట్ టైం	This is the first time to hear about Pichola but to see it	hate

Table 1: Sample Telugu text along with their English translations and corresponding labels

posed. The framework of the ML and TL models are shown in Figures 1 and 2. Pre-processing is the preliminary step in building learning models and it involves cleaning and transforming raw text data to a standardized format. Usually, text data contains noise in the form of: user mentions, hashtags, punctuation, digits, and hyperlinks, and eliminating this irrelevant information makes the data less complex and improves the performance of the classifier. Hence, in this work, punctuation, URLs, and stopwords are removed during pre-processing. Further, English stopwords available at NLTK library[2] and Telugu stopwords available at github[3] repository are used as references to remove English and Telugu stopwords from the given dataset. Further, the text in Roman script is converted to lowercase. The steps involved in building the proposed LR and Ensemble models are given below:

3.1 ML models

This section outlines the proposed LR and Ensemble models which are trained using feature vectors derived from n-grams of characters and words and sub-word tokens for identifying HOF content in code-mixed Telugu text and the steps are given below:

3.1.1 Feature Extraction

The role of feature extraction is to extract relevant features from the given data to train the learning models. Feature extraction techniques which are used to train LR and Ensemble models are described below:

- Character n-grams: are sequences of n consecutive characters. While one key stroke is enough to process each character in Roman script, characters in Indian languages like Telugu in its native script require more than one key stroke to process it. Therefore, in this work, to obtain character sequences for the given Telugu text where most of the text is in its native script, Telugu text is romanized using Indic transliterator[4] library. Subsequently, character n-grams in the range (1, 5) are obtained from the romanized Telugu text.

- Sub-word tokens: Sub-word tokenization algorithms prioritize breaking down rare words into smaller sub-word units, while leaving frequently used words (Bollegala et al., 2020). These algorithms are useful in representing both common and rare terms in a language. Therefore, this work utilizes Byte Pair Encoding algorithm to obtain sub-word tokens from the given Telugu text.

- Word n-grams: are sequences of 'n' consecutive words in a given text and these sequences capture the relationships between words. In this work, word sequences in the range (1, 3) are extracted from the given Telugu text.

The resultant character and word sequences and sub-words are vectorized using TFIDFVectorizer[5] and CountVectorizer[6] to construct the feature vectors.

3.1.2 Model Description

The proposed LR and Ensemble models are trained with the feature vectors obtained in the feature extraction step to classify the given code-mixed Telugu text as 'hate' or 'Non-hate' and description of each learning model is given below:

- **Logistic Regression (LR)** model: is used to predict the probability of certain classes based on dependent variables. The output of LR is always between (0 and 1), which is suitable for a binary classification task. Further, regularisation approaches in LR classifiers are useful for reducing overfitting in high dimensional space (Friedman et al., 2000).

[2] https://www.nltk.org/search.html?q=stopwords

[3] https://github.com/Xangis/extra-stopwords/blob/master/telugu

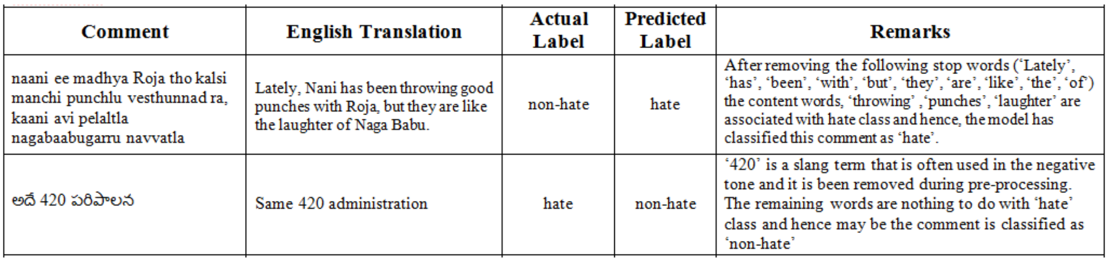
[4] https://github.com/libindic/indic-trans

[5] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[6] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

Models	Development set			Test set		
	Precision	Recall	Macro F1 score	Precision	Recall	Macro F1 score
LR model	0.76	0.76	**0.76**	0.65	0.65	**0.65**
Ensemble model	0.73	0.73	0.72	0.55	0.54	0.53
HateExplain_TL	0.72	0.72	0.72	0.49	0.49	0.48

Table 2: Performances of the proposed models



Comment	English Translation	Actual Label	Predicted Label	Remarks
naani ee madhya Roja tho kalsi manchi punchlu vesthunnad ra, kaani avi pelaltla nagabaabugarru navvatla	Lately, Nani has been throwing good punches with Roja, but they are like the laughter of Naga Babu.	non-hate	hate	After removing the following stop words (‘Lately’, ‘has’, ‘been’, ‘with’, ‘but’, ‘they’, ‘are’, ‘like’, ‘the’, ‘of’) the content words, ‘throwing’ ,‘punches’, ‘laughter’ are associated with hate class and hence, the model has classified this comment as ‘hate’.
అదే 420 పరిపాలన	Same 420 administration	hate	non-hate	‘420’ is a slang term that is often used in the negative tone and it is been removed during pre-processing. The remaining words are nothing to do with ‘hate’ class and hence may be the comment is classified as ‘non-hate’

Table 3: Samples of misclassification for code-mixed Telugu texts with respect to LR model

- **Ensemble model**: is a strategy for building a new classifier from several heterogeneous base classifiers taking benefit of the strength of one classifier to overcome the weakness of another classifier to get better performance for the classification task (Li et al., 2018). In this work, three ML classifiers (MNB, LR, and GNB) are ensembled with hard voting for identifying HOF content in code-mixed Telugu text. MNB is a probability-based ML classifier suitable for classification problems involving text data with discrete characteristics like word frequency counts (Ali et al., 2021). GNB is a probabilistic ML algorithm that relies on the Bayes theorem. By assuming feature independence, GNB determines the likelihood that a sample will fall into each of the predefined classes (Jain and Sharma).

3.2 HateExplain_TL model

TL is a technique within the broader field of ML that leverages knowledge gained from one task to improve the performance of a related task. It involves using pretrained models as a starting point and fine-tuning them for a specific task or domain (Hegde et al., 2023). The proposed HateExplain_TL model utilizes a HateExplainBERT[7] model pretrained on Twitter and Human Rationales text data that contains hatred or offensive texts exclusively making this model suitable for detecting HOF content. This BERT variant is fine-tuned on the pre-processed Train set and is used to train transformer classifier (ClassificationModel) to make the predictions.

4 Experiments and Results

The datasets provided by the shared task organizers for HOF content detection in Telugu code-mixed text consists of 2,061 samples belonging to ’Non-hate’ class and 1,939 samples belonging to ’hate’ class and 500 samples in the Test set. The sample code-mixed Telugu text, their English translations and the corresponding labels are shown in Table 1. Experiments are carried out, incorporating several feature combinations (sub-word count, word count, and character count), and classifiers (LR, SVM, k-Nearest Neighbors (k-NN), Ensemble (MNB, LR, and GNB), and HateExplain_TL). The models that showed considerable improvement on the Development set were subsequently tested on the Test set.

Predictions of the proposed models are evaluated based on macro F1 score and performances of the proposed models on Development and Test sets are shown in Table 2. The results reveal that LR model trained with TF-IDF of character and word sequences in the range (1, 5) and (1, 3) respectively, and sub-words, outperformed the other models with a macro F1 score of 0.65 securing 15th rank in the shared task. Few misclassified comments along with the actual and predicted labels (obtained from evaluating LR model on the given Test set) are shown in Table 3. It can be observed that most of the wrong classifications are due to removing stopwords and digits. Further, lack of context may also lead to misclassification in addition to rare

[7] https://huggingface.co/Hate-speech-CNERG/bert-base-uncased-hatexplain

words and wrong annotations.

5 Conclusion and Future Work

This paper describes the models submitted by our team - MUCS, to "Hate and Offensive Language Detection in Telugu Codemixed Text (HOLD-Telugu)" shared task at Dravidian-LangTech@EACL 2024, to identify HOF content in code-mixed Telugu text. Three models: i) LR model - a ML algorithm trained with TF-IDF of character and word sequences in the range (1, 5) and (1, 3) respectively and sub-words, ii) Ensemble model - a combination of ML classifiers (MNB, LR, and GNB) trained with CountVectorizer of character and word sequences in the range (1, 5) and (1, 3) respectively, iii) HateExplain_TL - a model based on TL approach with a BERT variant, are submitted to the shared task for detecting HOF content in Telugu code-mixed text. The proposed LR model outperformed the other models with a macro F1 score of 0.65 for Telugu code-mixed text. Effective feature extraction techniques and classifiers will be explored further.

References

Muhammad Z Ali, Sahar Rauf, Kashif Javed, Sarmad Hussain, et al. 2021. Improving Hate Speech Detection of Urdu Tweets Using Sentiment Analysis. In *IEEE Access*, volume 9, pages 84296–84305. IEEE.

Premjth B, Bharathi Raja, Prasanna Kumar Kumaresan, Saranya Rajiakodi, Sai Prashanth Karnati, Sai Rishith Reddy Mangamuru, and Janakiram Chandu. 2024. Findings of the shared task on hate and offensive language detection in telugu codemixed text (hold-telugu). In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, Malta. European Chapter of the Association for Computational Linguistics.

Somnath Banerjee, Maulindu Sarkar, Nancy Agrawal, Punyajoy Saha, and Mithun Das. 2021. Exploring Transformer Based Models to Identify Hate Speech and Offensive Content in English and Indo-Aryan Languages. In *arXiv preprint arXiv:2111.13974*.

Snehaan Bhawal, Pradeep Roy, and Abhinav Kumar. 2021. Hate Speech and Offensive Language Identification on Multilingual Code-mixed Text using BERT. In *Working Notes of FIRE 2021-Forum for Information Retrieval Evaluation (Online). CEUR*.

Danushka Bollegala, Ryuichi Kiryo, Kosuke Tsujino, and Haruki Yukawa. 2020. Language-Independent Tokenisation Rivals Language-Specific Tokenisation for Word Similarity Prediction. In *arXiv preprint arXiv:2002.11004*.

V Dikshitha Vani and B Bharathi. 2022. Hate Speech and Offensive Content Identification in Multiple Languages using machine learning algorithms. In *Forum for Information Retrieval Evaluation (Working Notes)(FIRE). CEUR-WS. org*.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2000. Additive Logistic Regression: A Statistical View of Boosting (With Discussion and a Rejoinder by the Authors). In *The annals of statistics*, volume 28, pages 337–407. Institute of Mathematical Statistics.

Asha Hegde, G Kavya, Sharal Coelho, and Hosahalli Lakshmaiah Shashirekha. 2023. MUCS@DravidianLangTech2023: Leveraging Learning Models to Identify Abusive Comments in Code-mixed Dravidian Languages. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 266–274.

Archika Jain and Sandhya Sharma. Hasoc19: Hate Speech Detection on Multimodal Dataset.

Ming Li, Peilun Xiao, and Ju Zhang. 2018. Text Classification based on Ensemble Extreme Learning Machine. In *arXiv preprint arXiv:1805.06525*.

Varsha Pathak, Manish Joshi, Prasad Joshi, Monica Mundada, and Tanmay Joshi. 2021. KBCNMUJAL@ HASOC-Dravidian-CodeMix-FIRE20: Using Machine Learning for Detection of Hate Speech and Offensive Code-mixed Social Media. In *arXiv preprint arXiv:2102.09866*.

Bharathi Raja andS Malliga andCN SUBALALITHA Priyadharshini, Ruba andChakravarthi, Premjith and-Murugappan Abirami S V, Kogilavani andB, and Prasanna Kumar Kumaresan. 2023a. Overview of shared-task on abusive comment detection in tamil and telugu. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, Varna, Bulgaria. Recent Advances in Natural Language Processing.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, S Malliga, Subalalitha Cn, SV Kogilavani, B Premjith, Abirami Murugappan, and Prasanna Kumar Kumaresan. 2023b. Overview of Shared-task on Abusive Comment Detection in Tamil and Telugu. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 80–87.

Pradeep Kumar Roy, Snehaan Bhawal, and Chinnaudayar Navaneethakrishnan Subalalitha. 2022. Hate Speech and Offensive Language Detection in Dravidian Languages using Deep Ensemble Framework. In *Computer Speech & Language*, volume 75, page 101386. Elsevier.

Abhishek Velankar, Hrushikesh Patil, Amol Gore, Shubham Salunke, and Raviraj Joshi. 2021. Hate and Offensive Speech Detection in Hindi and Marathi. In *arXiv preprint arXiv:2110.12200*.